ÉCOLE POLYTECHNIQUE DE L'UNIVERSITÉ DE NANTES
DÉPARTEMENT D'INFORMATIQUE

RAPPORT DE RECHERCHE ET DÉVELOPPEMENT

Fouille de données olfactives

Clustering de molécule odorantes par GNN (Graph Neural Networks)

Thomas CLOUET & Gabriel JOLLY

Février 2023

encadré par Fabrice GUILLET & Angélique VILLIÈRE

— Équipe Flaveur —
LABORATOIRE DE BIOCHIMIE D'ONIRIS
CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE

coordinateur : Philippe LERAY

Fouille de données olfactives

Clustering de molécule odorantes par GNN (Graph Neural Networks)

Thomas CLOUET & Gabriel JOLLY

Résumé

L'olfactométrie est une technique combinant l'analyse chimique et la perception humaine d'un juge entraîné (un nez). Elle permet, après une extraction de l'arôme d'un aliment, d'analyser et d'identifier, les molécules odorantes constituant cet arôme.
L'objectif de ce travail est de réduire la variabilité provenant de l'incertitude sur l'identification des odeurs et des molécules associées. Le travail consiste à appliquer des méthodes de réseaux de neurones pour les graphes (Graph Neural Networks ou GNN) sur les données afin d'extraire des sous-ensembles d'odeurs similaires.
Les travaux de l'équipe Brain de Google Research constituent l'amorçage principal de ce projet.

Remerciements

Nous souhaitons remercier M. Guillet et Mme. Villière pour leur encadrement et leurs retours tout au long de ce projet.

Table des matières

1 Introduction

L'introduction se divisera en différentes parties. Dans une première partie, nous détaillerons la problématique ainsi que les objectifs de ce projet. Puis, la seconde partie présentera la source originale sur laquelle nos travaux s'appuient. Pour finir, nous présenterons le plan de l'étude.

1.1 Présentation de la problématique et des objectifs

L'équipe « Flaveur » du laboratoire de biochimie d'Oniris est un des leaders européens dans l'étude de l'arôme des aliments. L'olfactométrie, technique utilisée dans ce laboratoire, combine l'analyse chimique et la perception humaine d'un juge entraîné (un nez). Elle permet, après une extraction de l'arôme d'un aliment, d'analyser et d'identifier, les molécules odorantes constituant cet arôme. Une des principales difficultés rencontrées, réside dans la forte variabilité des capacités de perception des juges (physiologique, anosmie, culturelle, représentation mentale des odeurs). Nous avons eu l'occasion de visiter le laboratoire et d'essayer l'olfactométrie. L'expérience était enrichissante, nous avons pu nous rendre compte à quel point il est difficile d'identifier et de nommer les odeurs associées aux molécules. En effet, si déterminer l'odeur d'un aliment ou d'une boisson peut paraître naturel, percevoir l'odeur des molécules odorantes de ces produits nécessite un véritable entraînement.

Par conséquent, l'objectif de ce projet est de réduire la variabilité provenant de l'incertitude sur l'identification des odeurs associées aux molécules. Pour répondre à cette problématique, nous construirons un modèle entraîné pour la perception des odeurs. Pour ce faire, nous avons accès à plusieurs sources de données :

— des données expérimentales en faible volume, récoltées à Oniris, où des juges identifient les odeurs détectées sur des produits ;
— une caractérisation par identification des molécules (numéro CAS) en fonction d'un étalon temporel et de l'odeur perçue ;

— les données des sites *the good scents company* [1] et *flavornet* [2] qui répertorient les associations molécule-odeurs communes sur de nombreux produits ;
— les données sur les propriétés physico-chimiques des molécules (site *PubChem* et *NIST*) ;
— un graphe de connaissances où les odeurs sont regroupées en pôles avec l'hypothèse qu'au sein d'un même pôle les odeurs se ressemblent davantage qu'entre 2 pôles distincts ;

1.2 L'équipe Brain de Google Research

L'équipe Brain est une unité interne de Google, consacrée à la recherche sur l'intelligence artificielle. En 2019, ils ont mené des recherches mélangeant les réseaux de neurones pour les graphes (GNN) et la prédiction des odeurs. Notre projet s'appuie sur leurs résultats, l'article scientifique *Machine Learning for Scent : Learning Generalizable Perceptual Representations of Small Molecules [Wil19b]* constitue notre première source. Dans ce document, l'équipe de Brain explique comment et pourquoi ils ont utilisé un modèle GNN pour la prédiction des odeurs.

1. http://www.thegoodscentscompany.com/index.html
2. https://www.flavornet.org/

1.3 Plan de l'étude

Le chapitre 2 introduit les GNNs au travers des articles [Wil19a] et [Wil19b]. Nous expliquerons l'origine, l'usage et la pertinence de ces modèles.

Le chapitre 3 étudie en profondeur le fonctionnement et l'application des GNNs. Pour cela, nous dresserons la liste des étapes nécessaires à la construction d'un GNN. De plus, les différents aspects mathématiques des couches du modèle seront expliqués.

Le chapitre 4 présente l'implémentation de notre solution sous la librairie *StellarGraph*. Nous expliquerons en détail toutes les parties de celle-ci, allant de la transformation des molécules sous la forme de graphe jusqu'à l'apprentissage de notre modèle. Enfin, nous présenterons nos résultats et pistes d'améliorations pour cette implémentation.

2

Les réseaux de neurones pour les graphes

Dans cette partie, nous allons nous appuyer sur les articles [Wil19a] et [Wil19b] pour expliquer l'origine et l'usage des GNNs.

2.1 Des méthodes classiques aux GNNs

Établir un lien entre la structure d'une molécule et ses potentielles odeurs est une tâche complexe. Ce problème est un challenge important dans le domaine de la chimie, la nutrition, la manufacture et l'environnement. Il existe plusieurs approches pour prédire les odeurs d'une molécule. Avant cela, il faut rendre la molécule « compréhensible » pour un réseau de neurones. En effet, il faut représenter les caractéristiques de la molécule sous la forme d'un vecteur. Pour ce faire, il existe diverses manières :

1. il est possible d'encoder les informations de la molécule de manière artisanale ;
2. on peut aussi s'appuyer sur la *Morgan fingerprints* :
   — les informations peuvent être encodées de manière binaire grâce à la *bit-based fingerprints (bFP)* ;
   — elles peuvent également être encodées de manière dénombrable avec la méthode *count-base fingerprints (cFP)* ;

Après avoir représenté la molécule sous la forme d'un vecteur, le modèle prédictif, tel qu'un random forest (RF) ou encore un k-nearest neighbor (KNN) sont maintenant dans la capacité de prédire ses potentielles odeurs.

Ces méthodes d'encodage de la molécule sous la forme d'un vecteur sont dites « classiques ». En effet, des méthodes plus complexes et personnalisables se démarquent. Les GNNs font leur apparition et sont conçus pour traiter des données structurées. Dans le cas de la prédiction des odeurs, les molécules peuvent être représentées sous forme de graphes, où les nœuds représentent les atomes et les arêtes représentent les liaisons entre ces atomes. De plus, les GNNs sont capables d'apprendre à partir de ces structures graphiques en utilisant des algorithmes pour propager l'information entre les nœuds du

graphe. Cela leur permet de prendre en compte les relations complexes entre les atomes et de fournir un encodage vectoriel précis de la molécule.

2.2 L'usage et la pertinence des GNNs

Dans la section précédente, nous avons introduit les capacités des GNNs, qui à partir d'un graphe représentant une molécule, peuvent produire en sortie un vecteur de taille fixe identifiant celle-ci, à la manière de la *Morgan fingerprints*. Ce vecteur peut ensuite être utilisé dans un réseau de neurones afin de déterminer les odeurs de la molécule.

Afin de mieux comprendre comment cela est possible, il faut observer le fonctionnement interne des GNNs. Leur architecture consiste en plusieurs couches d'échanges de messages. Ces couches successives permettent au fur et à mesure d'agréger les informations des atomes (les sommets du graphe) avec leurs voisins. La principale force de ces couches, réside dans la capacité à utiliser des informations de plus en plus profondes, comme sur le voisinage des voisins d'un sommet. La couche de sortie de ces échanges de messages est directement connectée à un perceptron multicouche pour la prédiction des odeurs (cf. figure 2.1).

Cet avantage suffit-il à se démarquer des méthodes d'encodage dites « classiques » ? Afin de répondre à cette question, nous pouvons nous baser sur les résultats de l'article [Wil19b]. En effet, dans ce document ont été effectué plusieurs prédictions d'odeurs sur des molécules

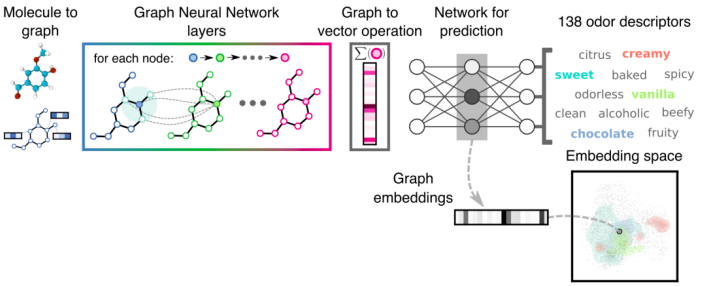

FIGURE 2.1 – Modèle de GNN utilisé pour l'analyse prédictive des odeurs sur les molécules [Wil19b]

en s'appuyant sur divers moyens d'encodage et modèles (cf. figure 2.2). Les méthodes suivantes ont été comparées :

- Mordred (librairie moléculaire, contenant sa propre fingerprint) ;
- Bit-based fingerprint (bFP) ;
- Count-based fingerprint (cFP) ;

Les modèles suivants ont été utilisés :

- Perceptron multicouche (MLP) ;
- Random forest (RF) ;
- K-nearest neighbor (KNN) ;

D'après ce tableau comparatif, le GNN possède le meilleur score AUROC, F1 ainsi que la meilleure précision, ce qui justifie son utilisation dans le cadre de la prédiction des odeurs.

Il est important de noter qu'il existe plusieurs types de GNNs qui varient en fonction de leur architecture. Dans

	AUROC	Precision	F1
GNN	**0.894 [0.888, 0.902]**	**0.379 [0.351, 0.398]**	**0.360 [0.337, 0.372]**
RF-Mordred	0.850 [0.838, 0.860]	0.311 [0.288, 0.333]	0.306 [0.283, 0.319]
RF-bFP	0.832 [0.821, 0.842]	0.321 [0.293, 0.339]	0.295 [0.272, 0.308]
RF-cFP	0.845 [0.835, 0.854]	0.315 [0.280, 0.332]	0.295 [0.272, 0.311]
KNN-bFP	0.791 [0.778, 0.803]	0.328 [0.305, 0.347]	0.323 [0.299, 0.335]
KNN-cFP	0.796 [0.785, 0.809]	0.333 [0.307, 0.351]	0.316 [0.292, 0.327]

FIGURE 2.2 – Comparatif des résultats de la prédiction des odeurs [Wil19b]

	GCN	MPNN
Message Passing Layers	concatenation message type, 4 layers of dim: [15,20,27,36], selu activation, max graph pooling	edge-conditioned matrix multiply message type, 5 layers of dim 43, GRU-update at each layer
Readout	Global sum pooling with softmax, 175 dim, one per MP layer and summed	Global sum pooling with softmax, 197 dim, one per MP layer with residual connections and summed
fully-connected neural net	2-layers of dim [96, 63] with relu, batchnorm, dropout of 0.47	3-layers of dim 392 with relu, batchnorm, dropout of 0.12 and l1/l2 regularization
Prediction	Multi-headed sigmoid, 138 tasks	
Training	Weighted-cross entropy loss, optimized with Adam, used learning rate decay with warm restarts, 300 epochs	

FIGURE 2.3 – Deux architectures de GNNs [Wil19b]

cet article, il est présenté deux GNNs distincts :

— Message Passing Neural Networks (MPNN) ;
— Graph Convolution Networks (GCN) ;

Ces deux variantes partagent un tronc commun, celui-ci consiste en des couches d'échange de messages, suivies d'un opérateur *reduce-sum*. Quant à la sortie de la dernière couche, elle est réutilisée au travers d'un perceptron multicouche (cf. figure 2.3).

	AUROC	Precision	Recall	F1
MPNN	0.890 [0.882, 0.898]	0.379 [0.352, 0.399]	0.387 [0.366, 0.408]	0.362 [0.335, 0.375]
GCN	0.894 [0.888, 0.902]	0.379 [0.351, 0.398]	0.390 [0.365, 0.412]	0.360 [0.337, 0.372]

FIGURE 2.4 – Les performances du MPNN et du GCN [Wil19b]

Dans cet article, des comparatifs entre ces deux GNNs ont été effectués. Grâce à cela, en plus de partager une structure très similaire, on peut remarquer qu'ils possèdent aussi des performances quasiment identiques (cf. figure 2.4).

La seule différence flagrante est que le GCN possède une architecture plus simple que le MPNN. Pour cette raison, l'équipe Brain de Google Research a décidé d'utiliser le modèle du GCN pour leurs recherches. Nous faisons le même choix et à partir de maintenant toute référence à un GNN correspondra à la structure du GCN proposé ci-dessus.

Nous allons donc nous appuyer sur la structure du GCN pour répondre au problème de la prédiction des odeurs. Dans la partie qui va suivre, nous expliquerons mathématiquement le fonctionnement du GCN.

3 Le fonctionnement et l'application des GNNs

Dans cette partie, nous allons expliquer l'architecture et le fonctionnement d'un GNN. Pour cela, nous allons nous appuyer sur un exemple d'application sur des molécules. Les articles [Ham20], [May20] et [KW17] nous ont permis de comprendre le fonctionnement des GNNs tandis que l'article [XYD+22] nous as permis de comprendre comment ils sont utilisés dans le cadre de la prédiction d'odeurs.

3.1 Représentation d'une molécule sous forme d'un graphe

Un GNN requiert un graphe en entrée. Dans notre cas, nous allons devoir transformer la molécule en un graphe. Pour cela, nous avons besoin de deux matrices :

— La matrice des caractéristiques : elle nous permet de représenter la molécule à l'aide de ses atomes. En effet, chaque ligne de la matrice correspond à un atome et contient des informations sur celui-ci.

— La matrice d'adjacence : elle nous permet de connaître les atomes connectés. Prenons A comme matrice d'adjacence. A est de dimension (n, n), avec n étant le nombre d'atomes. A_{ij} correspond à l'information de la connexion entre l'atome i et l'atome j. Si les deux atomes sont reliés, $A_{ij} = 1$, sinon $A_{ij} = 0$.

Penchons-nous maintenant sur la matrice des caractéristiques. N'ayant pas d'informations sur comment est encodé une molécule dans l'article [Wil19b], nous avons pris comme base l'article [XYD+22] pour l'encodage puis nous l'avons adapté pour qu'il soit plus pertinent. Cette matrice possède n vecteurs de taille 5, où n est le nombre d'atomes de la molécule. Chaque vecteur décrit des informations sur les atomes :

— Le symbole de l'atome (0 1 2 3 4 5 6) : (C O N S Cl Br H) ;

— Le degré de l'atome (0 1 2 3 4) : nombre de voisins de l'atome hors hydrogène ;

— La valence implicite (0 1 2 3 4) : nombre d'atomes d'hydrogène connectés à cet atome ;
— L'appartenance à un noyau aromatique (0 1) : faux ou vrai ;
— La chiralité de l'atome (0 1 2) : permet de savoir si le carbone est asymétrique, et donne une information sur sa position dans l'espace (0 : non-asymétrique, 1 : asymétrique et sens horaire, 2 : asymétrique et sens anti-horaire) ;

Par exemple, l'atome d'oxygène de la molécule d'eau (H_2O) sera représenté par le vecteur *[1, 0, 2, 0, 0]* (cf. figure 3.1).

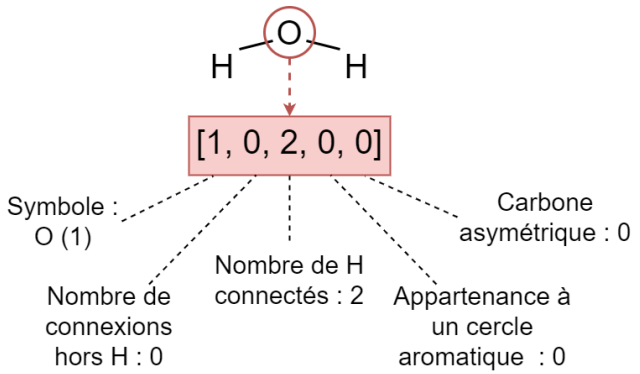

FIGURE 3.1 – Encodage de l'atome O de la molécule (H_2O)

3.2 Couches du GCN

Une fois la molécule transformée sous la forme d'une matrice, nous allons pouvoir l'utiliser comme entrée pour notre GCN. Après cela, les couches d'échanges de messages rentrent en actions.

Ces couches successives du GCN consistent à échanger les informations des nœuds entre leurs voisins. Chaque nœud agrège les informations de ses voisins ainsi que ses propres informations, puis change d'état pour la nouvelle étape (la couche suivante).

Prenons un exemple afin de mieux comprendre le fonctionnement de ces couches. Sur la figure 3.2 une molécule de vanilline est représentée. Nous allons simuler un échange de messages entre l'atome rouge et ses voisins, sur deux couches. Sur ce schéma, les documents incarnent les informations des atomes avant le processus d'échange. Lors de ce processus, l'atome rouge et ses voisins vont s'échanger leurs informations. On remarque qu'après la seconde couche, les informations des voisins sont agrégées aux informations d'origine de l'atome. Ces nouvelles informations seront ensuite utilisées pour les couches suivantes.

Nous allons maintenant expliquer mathématiquement le processus d'échange de messages. Afin de réaliser cette étape, il faut commencer par régulariser la matrice d'adjacence. Pour cela, il faut ajouter la matrice identité à la matrice d'adjacence, afin que les atomes s'incluent eux-mêmes dans le calcul avec leurs voisins. Après cela, il faut multiplier cette matrice avec la matrice des degrés. Cette dernière est diagonale et contient le nombre de voisins de chaque atome. Nous obtenons donc la formule suivante : $\hat{A} = D^{-1/2} * (A + I) * D^{-1/2}$, avec :

— $\hat{A}$: la matrice de régularisation ;

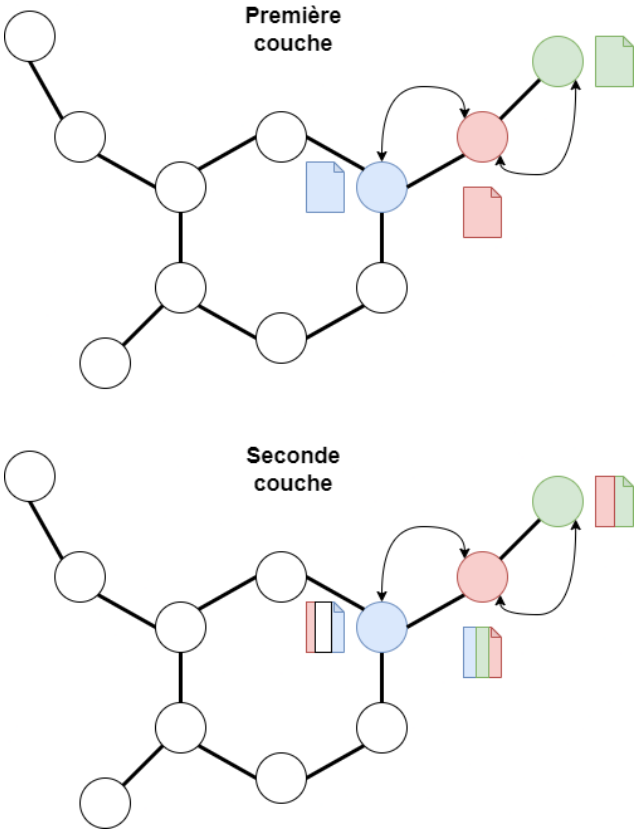

FIGURE 3.2 – Exemple d'un échange de messages entre l'atome rouge et ses voisins

— D : la matrice des degrés ;
— A : la matrice d'adjacence ;
— I : la matrice d'identité ;

La multiplication de matrice n'est pas commutative. Par conséquent, nous multiplions $(A + I)$ à droite et à gauche par $D^{-1/2}$ au lieu de simplement faire $(A+I)*D$. Une fois la matrice de régularisation $\hat{A}$ calculée, nous pouvons procéder à l'échange de messages. Pour cela, il suffit de multiplier $\hat{A}$ avec la matrice des caractéristiques de la couche d'entrée (matrice qui contient les informations des atomes de la molécule). Puis, il suffit de répéter le processus pour les couches suivantes : utiliser la matrice en sortie de ce calcul et la multiplier à nouveau avec $\hat{A}$. Pour chacune des couches du GCN, une matrice de poids W permet d'entraîner le GCN. Cette matrice est différente pour chaque couche (W^1, W^2, W^3, W^4). La taille de ces matrices varie en fonction de la dimension souhaitée à la couche suivante. De plus, s'ajoute à cela l'utilisation de la fonction d'activation SELU en sortie de chacune des couches. Pour conclure, on peut formuler le processus d'échange de messages de la manière suivante : $H^{l+1} = SELU(\hat{A} * H^l * W^l)$, avec :

— H^{l+1} : la matrice en sortie de la couche l ;
— H^l : la matrice en entrée de la couche l ;
— $\hat{A}$: la matrice de régularisation ;
— W^l : la matrice des poids pour la couche l ;
— SELU() : fonction d'activation SELU ;

La fonction SELU est donnée par :

$$f(x) = \lambda x \text{ si } x \geq 0$$

$$f(x) = \lambda\alpha(\exp(x) - 1) \text{ si } x < 0$$

avec $\alpha \approx 1.6733$ et $\lambda \approx 1.0507$.

Afin de résumer le calcul du processus d'échange de messages, nous pouvons nous appuyer sur la figure 3.3. Les diverses valeurs des matrices ont été recueillies à partir de la molécule de propanethiol (C_3H_8S). La matrice $\hat{A}$ est de dimension (4, 5). En effet, la molécule de propanethiol contient 4 atomes (hors hydrogène), et les informations de chacun de ses atomes sont encodés à l'aide d'un vecteur de taille 5. La dimension de la première couche du GCN est de 15. Par conséquent, la matrice en sortie de cette couche doit avoir pour dimension (4, 15). Pour ce faire, la matrice de poids aura pour dimension (5, 15).

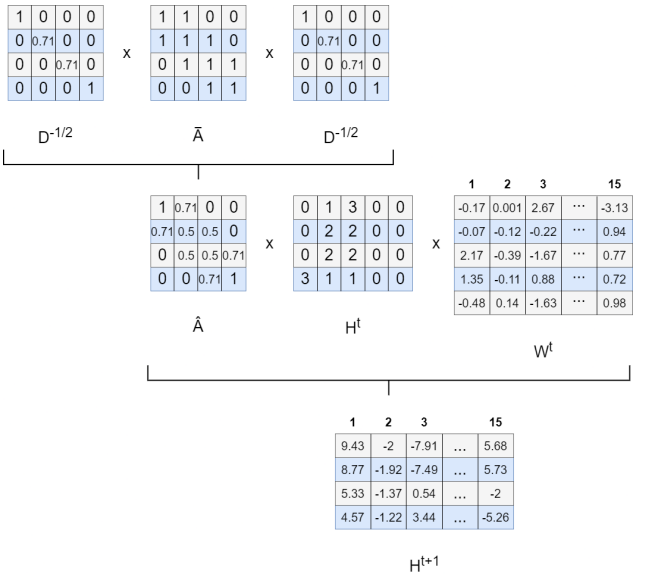

FIGURE 3.3 – Détail des calculs du processus d'échange de messages

3.3 Sortie du GCN

À la sortie de ces quatre couches, nous obtenons une matrice que nous allons transformer en un vecteur unique. Ce vecteur formera la représentation de la molécule en entrée, cet encodage peut être comparé à la *Morgan fingerprints*, la *bit-based fingerprints (bFP)* ou encore à la *count-base fingerprints (cFP)*.

Afin d'obtenir ce vecteur, un *global average pooling* sera appliqué à l'ensemble des sorties de chacune des couches du GCN. La figure 3.4 explique en détail le fonctionnement de cette méthode de pooling. Le vecteur de dimension 98 ainsi obtenu sera transformé en un nouveau vecteur de taille 175 grâce à une couche dense en appliquant la fonction d'activation softmax.

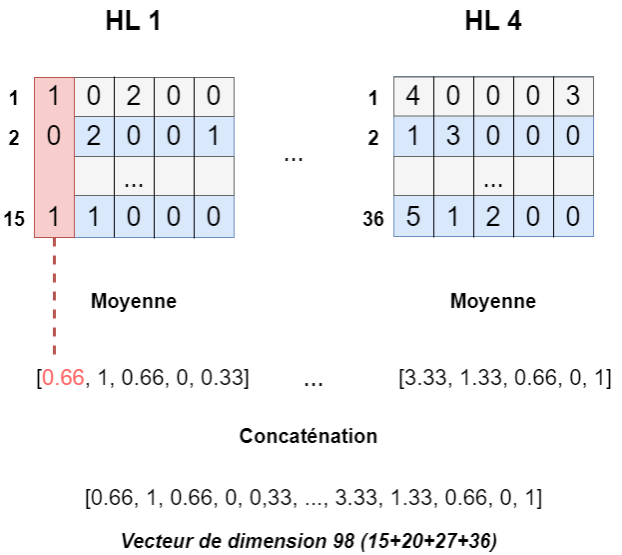

FIGURE 3.4 – Détail du fonctionnement du *global average pooling*

3.4 Prédicition des odeurs

Une fois notre vecteur final obtenu (*readout* du GCN), il ne nous reste plus qu'à le relier aux différents descripteurs d'odeurs. Pour cela, nous utilisons un réseau de neurones contenant deux couches entièrement connectées. À la sortie de ces deux couches, nous obtenons un vecteur de taille 98 qui correspond aux 98 descripteurs d'odeurs que nous avons sélectionnés. Chacune des valeurs du vecteur varie de 0 à 1, avec 0 représentant l'absence de l'odeur et 1 une probabilité de 100% de la présence de celle-ci.

3.5 Architecture complète du GNN

Nous nous appuyons sur l'architecture du GCN présentée en amont (cf. figure 2.3). Par conséquent, la dimension de la matrice en entrée est de (n, 5), où n étant le nombre d'atomes de la molécule.

Nous commençons par itérer sur les quatre couches du GCN en mélangeant la matrice des caractéristiques avec celle de régularisation. La taille de la matrice va évoluer entre chacune des couches : au départ, la dimension de la matrice sera (n, 15), puis (n, 20), puis (n, 27) et enfin (n, 36).

La matrice est ensuite transformée en un vecteur à l'aide d'un *global average pooling* et d'une fonction d'application softmax. Ce vecteur de taille 175 sera ensuite utilisé dans le réseau de neurones pour la prédiction des odeurs.

Ce réseau contient deux couches entièrement connec-

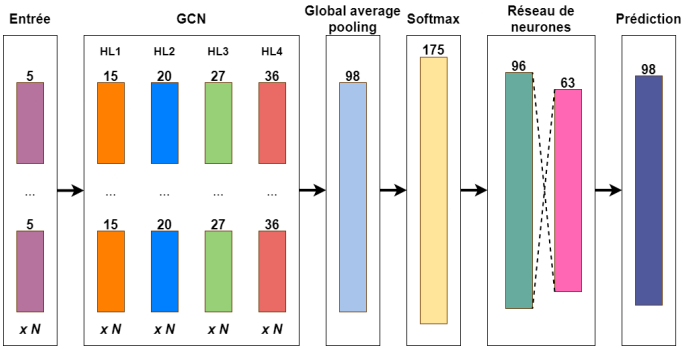

FIGURE 3.5 – Architecture du GCN

tées de taille 96 puis 63. Enfin, en sortie, nous obtenons un vecteur d'une taille égale au nombre de descripteurs d'odeurs à prédire, soit 98 (cf. figure 3.5).

4

Implémentation sous StellarGraph

4.1 Traitement des données d'entrée

Nous utilisons un jeu de données qui provient du laboratoire de biochimie d'Oniris. Celui-ci comporte 4 573 molécules. Chacune est décrite par son identifiant CAS et une suite de 0 et de 1 représentant l'absence ou la présence de 381 descripteurs d'odeurs.

Dans un premier temps, nous avons utilisé la librairie Cirpy [1] afin d'obtenir le code SMILE de chaque molécule. Nous nous sommes ensuite occupé de filtrer le jeu de données. Afin d'encoder les molécules en objet, on utilise la librairie RDKit [2]. Elle nous permet de construire un objet de type « molécule » à partir d'une formule SMILE. Nous avons donc commencé par retirer les molécules ne possédant pas de formule SMILE. De plus, comme dans l'article [Wil19b], nous avons défini un seuil minimal de présence de 30 occurrences pour chaque odeur afin que notre GCN s'entraîne sur des molécules dont les odeurs sont souvent présentes. Nous supprimons ensuite toutes les colonnes qui ne satisfont pas cette condition. Suite à cette suppression, nous nous retrouvons avec 98 descripteurs d'odeurs. On effectue un dernier traitement afin de retirer les molécules qui n'ont aucun descripteur d'odeur. Sachant que notre base de données n'est constituée que de molécules odorantes, cela veut dire qu'il manque des informations sur ces molécules. Une fois ce traitement fini, nous avons étudié notre jeu de données. Nous avons remarqué que l'atome de fluor et celui du sodium n'apparaissent qu'une seule fois dans tout notre jeu de données. Nous avons donc décidé de supprimer les 2 molécules contenant ces atomes afin de se concentrer sur des atomes plus fréquents. Nous obtenons donc un jeu de données comprenant 98 descripteurs d'odeurs et 2 843 molécules différentes.

Afin d'analyser nos résultats, nous avons fait un diagramme affichant la fréquence d'apparition d'une odeur dans notre jeu de données. Cette information est importante car elle impacte l'apprentissage du GCN (cf. fi-

1. https://pypi.org/project/CIRpy/
2. https://www.rdkit.org/

gure 4.1).

4.2 Implémentation

Lors du chapitre précédent, nous avons énuméré les différentes étapes permettant de construire un GCN. Nous allons maintenant voir un détail les outils qui nous aideront à l'implémentation d'un GCN en python. Vous pouvez retrouver le code sur notre Github [3].

Nous devons transformer la molécule en graphe et obtenir des informations sur celle-ci afin de construire la matrice de caractéristiques. Pour cela, nous allons utiliser *RDKit*. *RDKit* est une bibliothèque open source de chimio-informatique. Grâce à elle, nous pouvons facilement manipuler une molécule à partir de sa formule *SMILE*. Nous utiliserons les fonctions suivantes :

- *MolFromSmiles*, nous permet de récupérer une molécule à partir d'une formule *SMILE* ;
- *GetAdjacencyMatrix*, nous permet de récupérer la matrice d'adjacence ;
- *GetSymbol*, nous permet de récupérer le symbole d'un atome ;
- *GetDegree*, nous permet de récupérer le degré d'un atome ;
- *GetImplicitValence*, nous permet de récupérer la valence d'un atome ;
- *GetIsAromatic*, nous permet de récupérer l'appartenance à un noyau atomique ;

3. https://github.com/gabikun/PRED

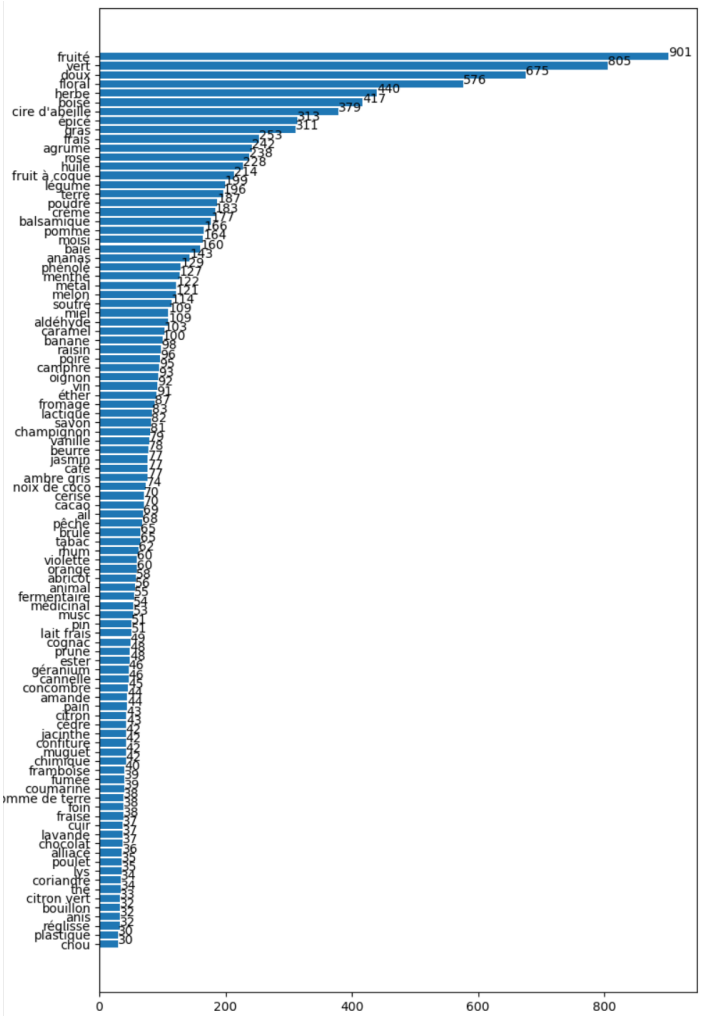

FIGURE 4.1 – Fréquence d'apparition des odeurs

— *GetChiralTag*, nous permet de savoir si l'atome est un carbone asymétrique ;

Pour la construction du GCN, nous utiliserons *StellarGraph*[4]. C'est une bibliothèque d'intelligence artificielle qui propose des algorithmes pour l'apprentissage automatique des graphes.

Nous avons aussi exploré divers projets *GitHub* contenant l'implémentation d'un GCN. Certains se basent sur *PyTorch*, tandis que d'autres n'utilisent que *Numpy* mais il nous semblait plus simple de faire notre GCN avec les outils proposés par *StellarGraph*.

Notre modèle est constitué de plusieurs étapes :

— La première est l'ensemble des 4 couches du GCN respectivement de taille 15, 20, 27 et 36. Chaque couche est suivie d'une fonction d'activation *SELU* ;

— En sortie de la 4ème couche, on retrouve une couche *GraphMeanPooling* qui permet de transformer nos N vecteurs de taille 36 en un seul vecteur de taille 175. Nous utilisons un *GraphMeanPooling* plutôt qu'un *GraphSumPooling* car cette couche n'existe plus dans *StellarGraph*. Elle existait dans une ancienne version, nous supposons donc qu'il est plus pertinent d'utiliser un *GraphMeanPooling* ;

— Une fois notre vecteur unique obtenu, nous pouvons procéder comme dans un réseau de neurones classique et passer au travers des couches entièrement connectées. Celles-ci sont de taille 96 puis 63 avec une fonction d'activation *RElu*. En sortie de chaque couche, nous précisons qu'il y a une couche de *BatchNormalisation* et une couche de *Dropout* pour que le modèle apprenne plus rapidement et soit plus stable ;

— Enfin, une dernière couche de taille 98 nous permet d'obtenir la prédiction de notre modèle sur les 98 descripteurs d'odeurs. Nous utilisons une fonction *sigmoïde* en sortie de chaque neurone ;

Une fois notre modèle créé, nous l'entraînons plusieurs fois et nous comparons les résultats afin de sélectionner le meilleur modèle. Lors de chaque essai, le modèle est entraîné 40 fois et nous sélectionnons celui qui a la meilleure précision.

Nous affichons une matrice de confusion afin de voir quelle odeur est prédite et nous faisons une réduction du nombre de dimensions grâce à l'algorithme t-SNE afin d'afficher les molécules sur un plan. Cela nous permet de voir des clusters dans la répartition des molécules.

4.3 Résultats

Comme dans l'article de Google, nous avons décidé d'utiliser la métrique de précision pour juger la performance de notre modèle. Nous avons 5 plis différents et 8 modèles différents pour chacun d'entre eux. Dans cet essai (cf. figure 4.2), notre meilleur modèle a eu un taux de précision de 65%. C'est ce modèle qui est utilisé dans la suite de nos résultats.

4. https://stellargraph.readthedocs.io/en/stable/

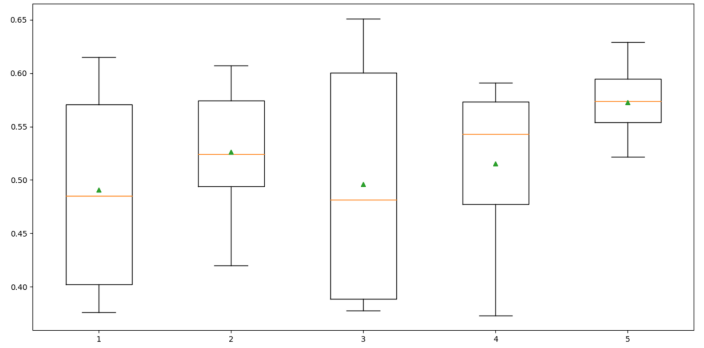

FIGURE 4.2 – Précisions des différents modèles en fonction du pli

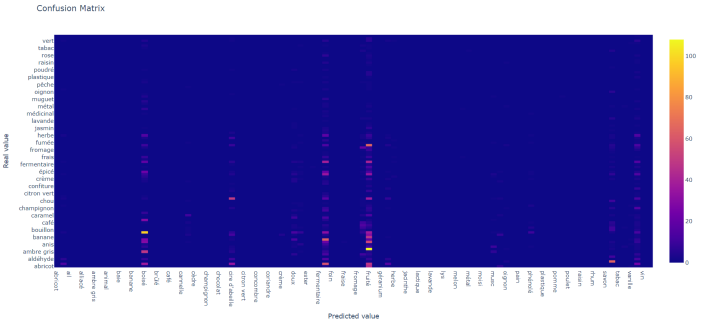

FIGURE 4.3 – Matrice de confusion des odeurs prédites et réelles

Nous avons affiché une matrice de confusion (cf. figure 4.3) afin d'afficher l'odeur que l'on prédit et l'odeur réelle. On remarque que notre modèle prédit un nombre d'odeurs très restreint. Cela s'explique par le fait que nous affichons que la meilleure précision dans la matrice de confusion.

Enfin, nous affichons un espace représentant les caractéristiques des molécules (cf. figure 4.4). Celui-ci nous permet de voir des apparitions de clusters. On peut remarquer que les clusters ont des similarités au niveau des couleurs, ce qui semble logique.

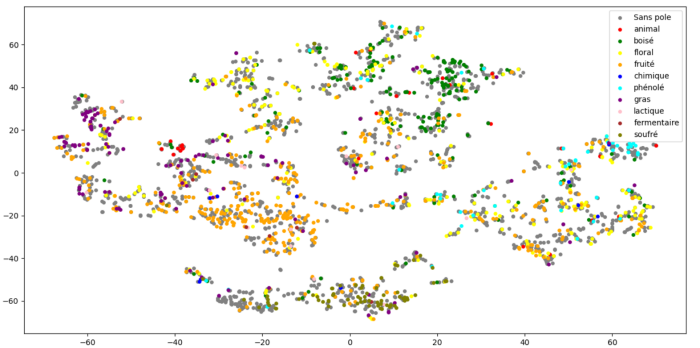

FIGURE 4.4 – Représentation en 2D des caractéristiques des molécules

4.4 Pistes d'améliorations

Il reste plusieurs points d'améliorations nécessaires pour avoir de meilleurs résultats :

— L'encodage des molécules : le type de liaison n'est pas encodé, il serait important de spécifier si la liaison est simple, double, triple... ;

— Faire une deuxième représentation des molécules avec les odeurs prédites et non les odeurs réelles afin de comparer les deux représentations entre elles ;

— Améliorer la matrice de confusion : Elle ne prend en compte que la meilleure prédiction pour chaque molécule alors qu'il y a plusieurs odeurs par molécule ;

— Remonter le GCN afin de trouver quelle sous-partie de la molécule est responsable d'une certaine odeur ;

5

Conclusion

Pour conclure, à travers les différents articles, nous avons déterminé que les GNNs pouvaient être utilisés pour prédire les odeurs émises par les molécules. En effet, ils surpassent les méthodes et les encodages classiques : *random forest (RF)*, *k-nearest neighbor (KNN)* et *bit-based fingerprints (bFP)*, *count-based fingerprints (cFP)*.

Grâce aux GNNs, nous pouvons obtenir un vecteur contenant toutes les informations de la molécule. Celui-ci, nous permet de déterminer les odeurs présentes dans la molécule. Pour cela, le GNN est suivi d'un réseau de neurones contenant des couches entièrement connectées. Il existe différentes architectures de GNNs. Dans notre cas, nous utilisons le *Graph Convolution Networks (GCN)*. Sa construction est simple et permet d'obtenir des résultats similaires aux *Message Passing Neural Networks (MPNN)*.

Ces premiers articles nous ont permis de mieux comprendre ce que sont les GNNs et leur utilité. Par la suite, nous nous sommes penchés sur la reproduction du GCN évoqué. En revanche, certaines informations nécessaires à sa construction sont absentes des documents. En effet, les caractéristiques des atomes et certains vocabulaires restent flous. Dans un second temps, nous nous sommes donc basés sur de nouvelles sources. Nous avons suivi les cours de Jure Leskovec [Les21] afin de mieux cerner le vocabulaire lié aux GNNs. De plus, nous avons pu éclaircir certaines zones d'ombre concernant le mécanisme interne du GCN : que représente la taille d'une couche ?

Grâce à ces travaux de recherche, nous avons été dans la capacité de reproduire et de construire notre GCN. Nous avons commencé à regarder les implémentations en Python. Différentes bibliothèques permettent de créer un GCN. Nous avons opté pour *Stellargraph*. Afin que le GCN soit fonctionnel, il faut transformer la molécule en graphe. Pour cela, nous nous aidons de *RDKit*. Cette bibliothèque permet de récupérer diverses informations sur la molécule telle que : sa matrice d'adjacence ou encore ses caractéristiques. Pour finir, nous avons pu construire notre GCN et le tester avec un jeu de données provenant

du laboratoire d'Oniris. Les différents résultats obtenus sont concluants, nous sommes dans la capacité de prédire les odeurs de molécules. Cependant, nous ne sommes pas encore dans la capacité de remonter le réseau et ainsi déterminer les sous-graphes des molécules responsables des odeurs.

Bibliographie

[Ham20] William L. Hamilton. *Graph Representation Learning*, volume 14. Morgan and Claypool, 2020. 11, 27

[KW17] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. ICLR, 2017. [1]. 11, 28

[Les21] Jure Leskovec. Machine learning with graphs, 2021. [2]. 22, 28

[May20] Inneke Mayachita. Understanding graph convolutional networks for node classification. 2020. [3]. 11, 27

[Wil19a] Alexander B. Wiltschko. Learning to smell : Using deep learning to predict the olfactory properties of molecules. 2019. [4]. 7, 8, 27

[Wil19b] Alexander B. Wiltschko. Machine learning for scent : Learning generalizable perceptual representations of small molecules. 2019. [5]. 7, 8, 9, 10, 11, 17, 25, 27

[XYD+22] iu Xiaofang, Cheng Yu, Luo Dehan, He Chunxia, K.Y. WONG Angus, and Liu Qi. Odor-gcn : Graph convolutional network for predicting odor impressions based on molecular structures. 2022. [6]. 11, 28

1. https://openreview.net/pdf?id=SJU4ayYgl
2. https://cs.stanford.edu/people/jure/teaching.html
3. https://towardsdatascience.com/understanding-graph-convolutional-networks-for-node-classification-a2bfdb7aba7b
4. https://ai.googleblog.com/2019/10/learning-to-smell-using-deep-learning.html
5. https://arxiv.org/pdf/1910.10685.pdf
6. https://assets.researchsquare.com/files/rs-1377643/v1_covered.pdf?c=1667972722

Table des figures

Liste des tableaux

Fiches de lecture

A.1 Machine Learning for Scent : Learning Generalizable Perceptual Representations of Small Molecules

Article [Wil19b], source originale de notre projet. Le document contient les recherches de l'équipe Brain de Google sur les méthodes de GNNs au sein de la prédiction des odeurs.

A.2 Learning to Smell : Using Deep Learning to Predict the Olfactory Properties of Molecules

Article [Wil19a], second document de l'équipe Brain de Google. Le fonctionnement des GNNs ainsi que leurs usages sont décortiqués au travers d'exemples sur les molécules.

A.3 Graph Representation Learning

Article [Ham20], contient diverses informations sur l'usage de l'intelligence artificelle dans le domaine de la recherche. L'explication de l'usage des GNNs est évoquée, un parallèle avec la problématique de la prédiction des odeurs est présent.

A.4 Understanding Graph Convolutional Networks for Node Classification

Article [May20], fait le lien entre les réseaux de neurones convolutifs et les réseaux de neurones pour les graphes. Le document, explique les similitudes et les différences entre ces deux méthodes.

A.5 Semi-supervised classification with graph convolutional networks

Article [KW17], explications mathématiques du fonctionnement des GCNs en classification supervisée.

A.6 Odor-GCN : Graph Convolutional Network for Predicting Odor Impressions Based on Molecular Structures

Article [XYD+22], démontre la pertinence des GCNs pour la prédiction d'odeurs en comparant avec d'autres méthodes.

A.7 Machine Learning with Graphs

Référence [Les21], contient les cours de Jure Leskovec sur les GNNs présentés à Stanford en 2021. Ces cours nous ont permis de comprendre des notions importantes des GNNs.

Planification

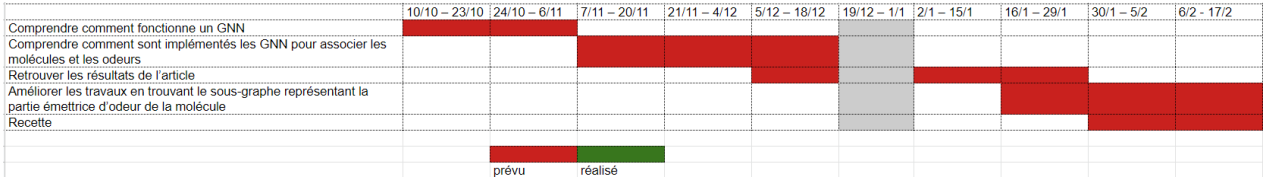

FIGURE B.1 – Planification prévisionnelle

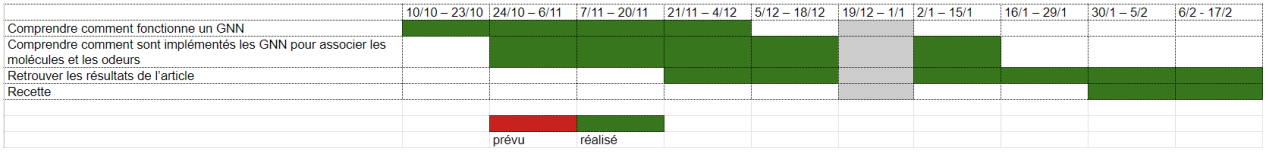

FIGURE B.2 – Planning effectif

Fiches de suivi

Cette annexe est *obligatoire*.

Fiche de suivi de la semaine 1
du 5 octobre 2022 au 11 octobre 2022

Temps de travail de Thomas CLOUET: 10 h 00 m
Temps de travail de Gabriel JOLLY: 10 h 00 m
Travail effectué.

- Lecture et compréhension de la problématique ainsi que des enjeux du sujet ;
- Visionnage de la vidéo expliquant l'olfactométrie et le rôle des molécules au sein des différentes odeurs ;
- Première lecture des deux références scientifiques qui nous ont été fournies ;

Échanges avec le commanditaire.

Lors de la réunion, nous avons expliqué avec nos mots ce que nous avions compris du sujet. Puis, il nous a été présenté les enjeux de celui-ci : les problèmes liés à l'identification d'une partie de la molécule, responsable d'une odeur. Afin de répondre à cette problématique, nous devrons nous appuyer sur des documents scientifiques afin de reproduire un GNN capable de déterminer quels sont les sous-graphes d'une molécule produisant une odeur.

Planification pour la semaine prochaine.

- Découverte du GNN, ce que c'est, en quoi il consiste et comment il peut être utilisé afin de répondre à la problématique ;

Fiche de suivi de la semaine 2
du 12 octobre 2022 au 18 octobre 2022

Temps de travail de Thomas CLOUET: 12 h 00 m
Temps de travail de Gabriel JOLLY: 12 h 00 m
Travail effectué.

— Approfondissement de la lecture des deux références scientifiques fournies.
— Visionnage de vidéos sur YouTube expliquant le fonctionnement d'un GNN.
— Comparaison des GNN avec les réseaux de neurones pour le traitement de l'image.

Travail non effectué.

— Travailler la compréhension du "coeur" du GNN : message passing layers.
— Se documenter à l'aide de nouvelles sources : Jure Leskovec.

Échanges avec le commanditaire.

Il existe deux façons différentes de représenter un GNN d'après l'un des articles : le MPNN et le GCN. Le GCN est celui qui est finalement retenu. En effet, sa structure est la plus simple d'accès et les performances entre ces deux approches sont similaires. Explication de la structure du GCN utilisé.

Planification pour la semaine prochaine.

— Comprendre les différents éléments qui composent le GCN.
— Se documenter sur les sources de données qu'ils utilisent dans l'article.

Fiche de suivi de la semaine 3
du 19 octobre 2022 au 25 octobre 2022

Temps de travail de Thomas CLOUET: 11 h 00 m
Temps de travail de Gabriel JOLLY: 11 h 00 m

Travail effectué.

— Poursuite de la compréhension de la structure du GNN : encodage seul des sommets, les informations sont représentées sous la forme d'un vecteur, l'objectif est de former un vecteur représentant la molécule (en agrégeant les informations des sommets au fil des couches).
— Explication du mécanisme du message passing : échange d'information entre les sommets voisins.
— Préparation de diverses questions sur le passage de la molécule en graphe : utilise-t-on le CAS ? Détermine-t-il diverses molécules ou variantes ? Détection des carbones asymétriques ?

Travail non effectué.

— Poursuivre les recherches sur l'encodage d'une molécule en graphe : quelles informations on représente ? Avec quelle source ?
— Récupérer tous les paramètres qui peuvent être utiles pour la réalisation du GCN.

Échanges avec le commanditaire.

Nous avons discuté de la méthode qui pourrait être utilisée afin de représenter une molécule sous la forme d'un graphe. Cela soulève la question des carbones asymétriques ? Le fingerprint, qu'est-ce que c'est ? Peut-il déterminer le cas des carbones asymétriques ?

Planification pour la semaine prochaine.

— Recherche sur la façon de passer d'une molécule à un graphe (fingerprint, smile) et sur les informations à encoder.

— Se documenter sur les sources de données qu'ils utilisent dans l'article.

Fiche de suivi de la semaine 4
du 2 novembre 2022 au 8 novembre 2022

Temps de travail de Thomas CLOUET: 10 h 00 m
Temps de travail de Gabriel JOLLY: 10 h 00 m

Travail effectué.

— Détermination des différentes sources de données qui ont été utilisées dans l'article : LeffingWell PMP 2001, GoodScents perfume et la librairie Mordred.
— Distinction entre bit-based fingerprint et count-based fingerprint.
— Meilleure compréhension du GNN, son objectif est d'obtenir un vecteur décrivant la molécule (à la façon d'un fingerprint).
— Recherche sur les informations qui constituent les vecteurs des sommets du graphe.

Travail non effectué.

— Proposer l'architecture complète du GCN : fonctionnement, couches, paramètres.
— Poursuivre les cours de J. Leskovec sur les GNN et GCN.

Échanges avec le commanditaire.

Nous avons discuté des données qui sont utilisées dans l'article. Ils utilisent majoritairement des molécules odorantes, ce qui pourrait biaiser l'apprentissage ? La librairie Mordred permet d'obtenir des informations complémentaires sur les molécules. Nous avons compris pourquoi les méthodes d'encodage d'une molécule sont comparés au GNN, car celui-ci permet d'obtenir aussi une représentation de la molécule (sous la forme d'un vecteur). Cependant, il reste des zones d'ombres, notamment l'usage des dimensions au sein du GCN : que représentent-elles ? Qu'est-ce que le max graph pooling, la fonction d'activation SELU ? Quelle est la fonction d'agrégation du message passing ?

Planification pour la semaine prochaine.

Réalisation d'un diagramme expliquant le fonctionnement du GCN et à quoi servent ses différents composants ainsi que leurs paramètres.

Fiche de suivi de la semaine 5
du 9 novembre 2022 au 15 novembre 2022

Temps de travail de Thomas CLOUET: 10 h 00 m
Temps de travail de Gabriel JOLLY: 10 h 00 m

Travail effectué.

— Premier essai pour la modélisation complète du GCN.

— Explication mathématique du message passing : cours de J. Leskovec.
— Présentation des fonctions d'activation : SELU / RELU .
— Recherches sur ce qu'est le "max graph pooling".
— Schémas explicatifs sur la partie fully-connected layers.

Travail non effectué.

— Expliciter les formules mathématiques sous la forme de dimensions : vecteurs / matrices...
— Éclaircir la partie convolution : à quoi correspondent les dimensions ?
— Trouver des exemples concrets de GCN.

Échanges avec le commanditaire.

Présentation de l'architecture du GCN. Il faut retravailler la partie message passing en rendant les slides plus concrètes. La partie convolution reste toujours un mystère : s'agit-il vraiment des dimensions des vecteurs ? Il faut déterminer ce qu'est le "max graph pooling" et comprendre où il est utilisé. En effet, ici les dimensions augmentent, cependant l'objectif du pooling est de réduire.

Planification pour la semaine prochaine.

— Trouver la réponse concernant les différentes couches de dimensions [15, 20, 27, 36].
— Trouver où est utilisé le max graph pooling et sa fonction.

Fiche de suivi de la semaine 6
du 16 novembre 2022 au 22 novembre 2022

Temps de travail de Thomas CLOUET: 9 h 00 m

Temps de travail de Gabriel JOLLY: 9 h 00 m

Travail effectué.

— Second essai pour la Modélisation complète du GCN.
— Explication de la partie "représentation de la molécule en graphe" sous la forme de matrices.
— Recherches sur le "max graph pooling".

Travail non effectué.

— Compréhension de la partie couches du GCN (dimensions, ...).
— Trouver le calcul de la matrice des poids.

Échanges avec le commanditaire.

Partie message passing expliquée en grande partie. La partie convolution reste incertaine tout comme le "max graph pooling". Comme nous commençons à tourner en rond, nous allons commencer à développer un GCN simple. Celui-ci nous permettra de comprendre davantage le mécanisme des GCNs. De plus, la partie "représentation de la molécule en graphe" sera déjà faite.

Planification pour la semaine prochaine.

— Développer un premier exemple de GCN très simple afin d'avoir les bases.
— Développer la transformation de la molécule en graphe puis en matrices.

Fiche de suivi de la semaine 7
du 23 novembre 2022 au 29 novembre 2022

Temps de travail de Thomas CLOUET: 15 h 00 m
Temps de travail de Gabriel JOLLY: 15 h 00 m

Travail effectué.

- Mise en place du GCN sur python.
- Utiliser le SMILE d'une molécule afin de récupérer ses informations grâce à RDKit (matrice d'adjacence, features, ...)
- Création d'un GCN avec numpy afin de comprendre le fonctionnement des couches de message passing.
- Prise en main du GCN avec PyTorch.
- Recherches sur la façon de réaliser un GCN en python.

Travail non effectué.

- S'occuper de l'entrée : récupérer et traiter plusieurs molécules. (voir avec DeepChem.io)
- Implémenter la suite du GCN : le readout, les full connected layers et le training.

Échanges avec le commanditaire.

Lors de la réunion nous avons présenté nos scripts python. Nous avons détaillé comment nous récupérons les informations de la molécules via le SMILE. Puis nous avons expliqué en détail les calculs réalisés dans une couche du GCN. Après cela, nous avons présenté notre modèle de GCN. Nous avons aussi affirmé que les différentes couches représentent bien la taille d'un vecteur décrivant un sommet. A la fin du GCN, nous obtenons donc une matrice contenant N vecteurs de taille 36, avec N le nombre de sommets de la molécule. La question est maintenant, comment passer ce N*36 en un vecteur de taille 175 qui sera utilisé par le réseau de neurones.

Planification pour la semaine prochaine.

- Préparer la soutenance.
- Avoir une entrée des données propres afin d'obtenir les matrices d'adjacence et des features définitives.
- Déterminer si la matrice des poids est globale ou pour chaque bloc.
- Déterminer ce qu'est le graph sum pooling pour chaque couche cachée.

Fiche de suivi de la semaine 8
du 30 novembre 2022 au 6 décembre 2022

Temps de travail de Thomas CLOUET: 15 h 00 m
Temps de travail de Gabriel JOLLY: 15 h 00 m

Travail effectué.

- Préparation de la soutenance.
- Première version du rapport.

Travail non effectué.

- S'occuper de l'entrée : récupérer et traiter plusieurs molécules. (voir avec DeepChem.io)
- Implémenter la suite du GCN : le readout, les full connected layers et le training.

— Avoir une entrée des données propres afin d'obtenir les matrices d'adjacence et des features définitives.
— Déterminer si la matrice des poids est globale ou pour chaque bloc.
— Déterminer ce qu'est le graph sum pooling pour chaque couche cachée.

Échanges avec le commanditaire.

Nous devons faire attention à nos citations, expliquer clairement l'objectif du projet en présentant l'article ainsi que les données que nous possédons en entrée et en sortie.

Planification pour la semaine prochaine.

— Avoir une entrée des données propres afin d'obtenir les matrices d'adjacence et des features définitives.
— Réimplémenter le GCN avec DeepChem.
— Continuer les parties manquantes (readout, full connected layers, training).

Fiche de suivi de la semaine 9
du 7 décembre 2022 au 13 décembre 2022

Temps de travail de Thomas CLOUET: 10 h 00 m
Temps de travail de Gabriel JOLLY: 10 h 00 m

Travail effectué.

— Lecture de la documentation de DeepChem.
— Recherches d'exemples de GCN avec DeepChem sur git.
— Création de notre GCN à l'aide de DeepChem et des ressources sur git.

Travail non effectué.

— Retravailler l'entrée des données, il faut adapter the good scents company.
— Déterminer ce que fait un max graph pooling et son équivalent en DeepChem et l'implémenter.
— Faire de même pour le global sum pooling et comprendre comment il fonction sur des couches de taille différentes.
— Créer le réseau de neurones utilisé pour la prédiction des odeurs.

Échanges avec le commanditaire.

Lors de cette réunion, nous avons présenté notre passage sur DeepChem. Nous avons évoqué les différentes fonctions que nous avons utilisées afin de retranscrire notre ancien code. Après cela, nous avons discuté d'un problème : les données utilisées par DeepChem ont un format spécifique adapté à leur base de données. Or, nous ne comptons pas utiliser leur base. Nous devons alors trouver un moyen pour adapter les données de The Good Scents Company au format des données de DeepChem.

Planification pour la semaine prochaine.

— Présenter l'architecture finale en DeepChem du GCN.
— Montrer des résultats à l'aide des bases de DeepChem.

Fiche de suivi de la semaine 10
du 14 décembre 2022 au 3 janvier 2023

Temps de travail de Thomas CLOUET: 10 h 00 m

Temps de travail de Gabriel JOLLY: 10 h 00 m

Travail effectué.

— Implémentation du GCN avec DeepChem.
— Implémentation des couches entièrement connectées en PyTorch.

Travail non effectué.

— Retravailler l'entrée des données.
— Refaire les couches entièrement connectées en DeepChem.
— Assembler tout le GCN.

Échanges avec le commanditaire.

Lors de cette réunion, nous avons présenté notre début d'implémentation sur DeepChem. Nous nous sommes focalisé sur la partie GCN et les paramètres que nous utilisons en faisant le parallèle avec les données de l'article. Nous changerons les données d'entrées une fois que le GCN sera totalement assemblé et prêt. Nous pouvons tester avec les données proposées par les bases de données de DeepChem pour vérifier que notre GCN fonctionne.

Planification pour la semaine prochaine.

— Refaire les couches entièrement connectées en DeepChem.
— Assembler tout le GCN.

Fiche de suivi de la semaine 11
du 4 janvier 2023 au 10 janvier 2023

Temps de travail de Thomas CLOUET: 9 h 00 m

Temps de travail de Gabriel JOLLY: 9 h 00 m

Travail effectué.

— Transformation des couches totalement connectées en Keras.
— Assemblage de la partie GCN + Réseau de neurones entièrement connecté.

Travail non effectué.

— Retravailler l'entrée des données.
— Refaire le GCN sous Stellar Graph afin de comprendre les derniers rouages du GCN.

Échanges avec le commanditaire.

Lors de cette réunion nous avons évoqué nos difficultés à faire fonctionner notre GCN et la partie réseau de neurones sous deepchem. En effet, certaines de nos fonctions surchargées ne sont pas prises en compte lors de l'apprentissage du modèle. Deepchem comporte trop de "boîtes noires" il est donc difficile de comprendre le bon fonctionnement du modèle. Nous allons donc refaire ce GCN sous Stellar Graph, où cette fois-ci nous avons des exemples de GCN et de réseau de neurones assemblés.

Planification pour la semaine prochaine.

— Faire l'implémentation du GCN en StellarGraph pour comprendre.
— Transposer cette implémentation sur Deepchem.

Fiche de suivi de la semaine 12
du 11 janvier 2023 au 17 janvier 2023

Temps de travail de Thomas CLOUET: 11 h 00 m

Temps de travail de Gabriel JOLLY: 11 h 00 m

Travail effectué.

- Implémentation du GCN en StellarGraph.
- Transformation des données d'entrées.

Travail non effectué.

- Utiliser le vrai jeu de données et bien le formater.
- Présenter des résultats.

Échanges avec le commanditaire.

Lors de cette réunion nous avons présenté notre implémentation du GCN sous Stellargraph. Cela comprend, la partie échange de messages et le réseau de neurones en sortie contenant des couches entièrement connectées. Après cela, nous avons montré comment nous avons adapté l'entrée du GCN afin de pouvoir utiliser notre jeu de données personnalisé. M. Guillet nous a fourni un jeu de données plus conséquent afin que nous puissions produire nos premiers résultats.

Planification pour la semaine prochaine.

- Modifier l'entrée pour faire correspondre le format des données avec le format nécessaire pour le GCN.
- Transposer l'implémentation de Stellargraph sur Deepchem.

Fiche de suivi de la semaine 13
du 18 janvier 2023 au 24 janvier 2023

Temps de travail de Thomas CLOUET: 20 h 00 m

Temps de travail de Gabriel JOLLY: 20 h 00 m

Travail effectué.

- Formatage du CSV et nettoyage des données.
- Ajout de la prise en compte des carbones asymétriques.
- Filtrage des données (on garde les odeurs qui sont présentes au moins 10 fois).
- Visualisation boxplot des performances du modèle pour chaque plis de la validation croisée.
- Matrice de confusion (odeurs prédites / odeurs réelles).
- TSNE (space embedding) des molécules.

Travail non effectué.

- Passer à 30 occurrences minimum pour la présence des odeurs.
- Changer les paramètres pour se mettre dans le cadre du papier.
- Ajouter la métrique "précision" en plus de "accuracy".
- Créer un histogramme des fréquences des odeurs.
- Peut-on encoder le type des liaisons et comment les utiliser ?.
- En quoi le GCN de stellargraph est-il supervisé ?.

Échanges avec le commanditaire.

Nous avons eu deux réunions cette semaine : mardi et vendredi. Lors de la première réunion nous avons montré

le bon fonctionnement de notre GCN sous stellargraph et la précision du modèle. Lors de la seconde réunion, nous avons présenté des visualisations permettant de mieux comprendre les résultats de notre modèle.

Planification pour la semaine prochaine.

Nous allons remplacer l'ensemble de nos paramètres pour qu'ils soient similaires à ceux utilisés par le papier. Puis nous compléterons les visualisations (TSNE et histogramme des fréquences). Pour finir, nous recherchons les réponses aux quelques questions restantes.

Fiche de suivi de la semaine 14
du 25 janvier 2023 au 31 janvier 2023

Temps de travail de Thomas CLOUET: 10 h 00 m
Temps de travail de Gabriel JOLLY: 10 h 00 m

Travail effectué.

- Filtrage des données (on garde les odeurs qui sont présentes au moins 30 fois).
- Histogramme de fréquence d'apparition des odeurs.
- Ajout de batch normalization et de dropout dans la partie réseau de neurones.
- TSNE avec une couleur pour chaque pôle.
- Changement des metrics : "Précision" et "AUROC".

Travail non effectué.

- Un TSNE avec seulement les pôles "non duo".
- Un autre TSNE avec des couleurs pour les bonnes prédictions.
- faire tourner avec le fichier des odeurs présent 10 fois minimum et comparer avec le seuil des 30 présence d'odeurs.

Échanges avec le commanditaire.

Lors de la réunion nous avons présenté nos nouveaux résultats avec la métrique "précision". Les résultats se rapprochent du papier original. De plus, notre TSNE a du sens dans sa répartition spatiale des molécules, cependant, il faut modifier les couleurs car actuellement les molécules bi-pôles ne sont pas prises en compte.

Planification pour la semaine prochaine.

Écriture du rapport et création des 2 TSNE. S'il nous reste du temps, nous ferons une comparaison avec le fichier des odeurs présentes au moins 10 fois.

Fiche de suivi de la semaine 15
du 1 février 2023 au 7 février 2023

Temps de travail de Thomas CLOUET: 10 h 00 m
Temps de travail de Gabriel JOLLY: 10 h 00 m

Travail effectué.

- Ecriture du rapport final.

— Visite du laboratoire d'Oniris.
Travail non effectué.
—
Échanges avec le commanditaire.
Planification pour la semaine prochaine.
—

Fiche de suivi de la semaine 16
du 8 février 2023 au 14 février 2023

Temps de travail de Thomas CLOUET: 10 h 00 m
Temps de travail de Gabriel JOLLY: 10 h 00 m
Travail effectué.
— Prépration de la soutenance finale
Travail non effectué.
—
Échanges avec le commanditaire.
Planification pour la semaine prochaine.
—

Le tableau C.1 récapitule le taux d'avancement du projet. Rappelons que le temps de travail théorique *minimal* correspond au temps indiqué sur la maquette pédagogique auquel on ajoute un strict minimum de 20 % correspondant au travail personnel hors emploi du temps. La partie « haute » de la fourchette correspond à 50 % de temps supplémentaire au titre du travail personnel.

	Temps prévu		Thomas CLOUET			Gabriel JOLLY		
	bas	haut	hebdo.	Σ	%	hebdo.	Σ	%
Semaine	h : m	h : m	h : m	h : m		h : m	h : m	
1	10 : 00	12 : 30	10 : 00	10 : 00	100 (80)	10 : 00	10 : 00	100 (80)
2	20 : 00	25 : 00	12 : 00	22 : 00	110 (88)	12 : 00	22 : 00	110 (88)
3	30 : 00	37 : 30	11 : 00	33 : 00	110 (88)	11 : 00	33 : 00	110 (88)
4	40 : 00	50 : 00	10 : 00	43 : 00	107 (86)	10 : 00	43 : 00	107 (86)
5	50 : 00	62 : 30	10 : 00	53 : 00	106 (84)	10 : 00	53 : 00	106 (84)
6	60 : 00	75 : 00	9 : 00	62 : 00	103 (82)	9 : 00	62 : 00	103 (82)
7	70 : 00	87 : 30	15 : 00	77 : 00	110 (88)	15 : 00	77 : 00	110 (88)
8	80 : 00	100 : 00	15 : 00	92 : 00	115 (92)	15 : 00	92 : 00	115 (92)
9	90 : 00	112 : 30	10 : 00	102 : 00	113 (90)	10 : 00	102 : 00	113 (90)
10	100 : 00	125 : 00	10 : 00	112 : 00	112 (89)	10 : 00	112 : 00	112 (89)
11	110 : 00	137 : 30	9 : 00	121 : 00	110 (88)	9 : 00	121 : 00	110 (88)
12	120 : 00	150 : 00	11 : 00	132 : 00	110 (88)	11 : 00	132 : 00	110 (88)
13	130 : 00	162 : 30	20 : 00	152 : 00	116 (93)	20 : 00	152 : 00	116 (93)
14	140 : 00	175 : 00	10 : 00	162 : 00	115 (92)	10 : 00	162 : 00	115 (92)
15	150 : 00	187 : 30	10 : 00	172 : 00	114 (91)	10 : 00	172 : 00	114 (91)
16	160 : 00	200 : 00	10 : 00	182 : 00	113 (91)	10 : 00	182 : 00	113 (91)

TABLE C.1 – Avancement du projet par rapport au temps de travail théorique minimal (respectivement haut)